

## 周报

前面几周一直在写浮点时序体数据压缩域体绘制(CDVR)的论文以及同步的代码实现。

图 1 是压缩端的流程图

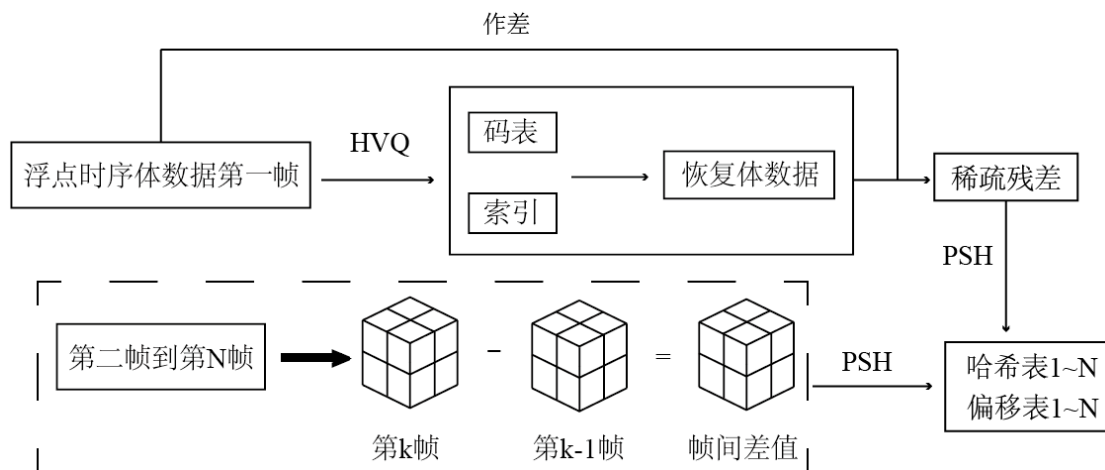


图 1 压缩端流程图

压缩端主要包括层次矢量量化压缩(HVQ)和完美空间哈希压缩(PSH)，流水线如图 1 所示，假设数据有  $N$  帧，我们将浮点时序体数据分成第一帧和其余帧两个部分。首先，使用 HVQ 错误!未找到引用源。对第一帧进行矢量量化压缩，得到码表和索引数据；然后在 CPU 端解压缩得到该帧的恢复体数据；再将原始数据和恢复体数据作差得到稀疏的残差体数据，应用 PSH 压缩该残差，得到第一个哈希表和偏移表；对于第二帧到第  $N$  帧，依次将其与前一帧两两作差得到  $N-1$  个帧间差值体数据。由于数据的时空连贯性，这些帧间差值体数据是稀疏的，即可利用 PSH 压缩这些帧间差值体数据，获

得另外  $N-1$  个哈希表和偏移表。

图 2 是解码端和绘制的流程图。解压缩端包括基于 GPU 的实时解压缩与直接体绘制，在进行时序体数据的压缩域体绘制时，对于第一帧，我们将压缩端获得的矢量量化码表、索引数据、哈希表和偏移表构建成纹理并载入 GPU。绘制时，对码表纹理和索引体纹理进行纹理查询，重构第一帧体数据，然后对哈希表纹理和偏移表纹理进行纹理查询获取残差数据，叠加到对应的体素中，即可获得第一帧的近似无损的重构数据，以传统体绘制流程绘制此恢复体数据。对于后续帧，只需要对对应的哈希表纹理和偏移表纹理进行纹理查询，获取帧间差值数据并叠加到前一帧重构的体数据上即可获得对应帧的近似无损的重构体数据值，然后进一步进行体绘制

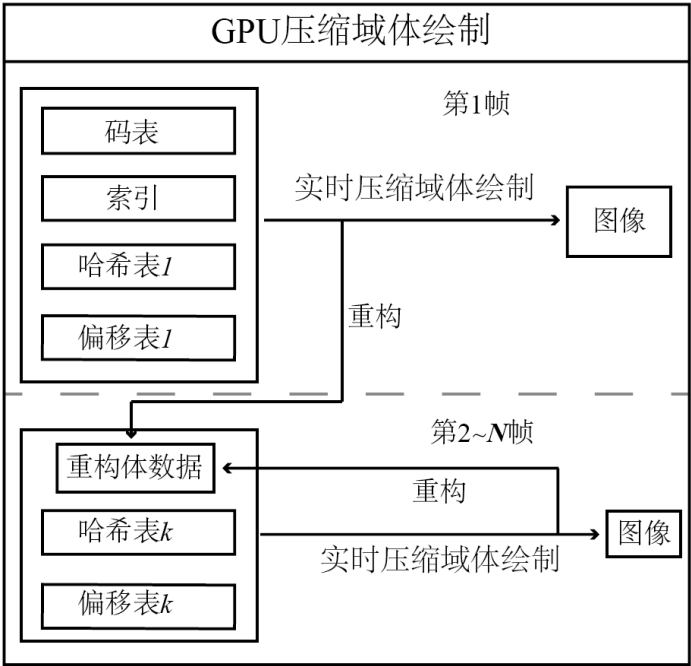


图 2 解码端和绘制流程图

图 3 是 HVQ 压缩端的流程图。对于一个浮点时序体数据的第一帧，我们首先将其分成  $4*4*4$  大小的数据块（矢量数据维度为 64）并对其编码、构造矢量，然后对每一个数据块进行拉普拉斯分解，将块内数据分解为三个频率波段，进行分层表示。三个层次的分辨率分别为  $4*4*4$ ,  $2*2*2$  和  $1*1*1$ 。数据值如下计算：首先计算  $4*4*4$  的原始数据块的均值  $M_0$  并将之分为 8 个  $2*2*2$  的小数据块，分别计算小数据块的均值  $M_1 \sim M_8$ 。原始数据的 64 个数据值与其各自所在小块均值之差  $M_i (i \in [1,8])$  构成了第一层次的 64 维数据， $M_1 \sim M_8$  与  $M_0$  之差构成了第二层次的 8 维数据， $M_0$  构成第三层数据。最终，每个  $4*4*4$  的原始数据块可以用分解完成的三层数据表示。对于第一和第二层次的数据，用矢量量化进行压缩，设定码表大小为 256，索引数据类型为无符号字符型，每个索引大小为 8 比特，矢量的数据维度分别为 64 和 8。由于 LBG 算法对于初始码表比较敏感，因而 HVQ 采用改进的基于主元分析的分裂策略训练初始码表，然后进行矢量量化。第三层的数据是原始数据块的平均值，直接作为对应层次的索引数据。如图 3 中部所示， $4*4*4$  的浮点型数据块最终可以用两个索引值和一个均值替代，两个索引值所索引的码元大小分别是 64，8，索引数据的和均值的大小都为原始数据块大小的  $1/256$ 。在具体的实现过程中，上述三个数据被构造成一个三通道的三维索引纹理，两个层次的码表被构造成两个二

维的单通道码表纹理，如图 3 右端所示。目前，对于 HVQ 的压缩以及压缩域体绘制已经有结果，它也可以被直接应用到气象数据的压缩中。浮点体数据压缩率在 40-80 之间。

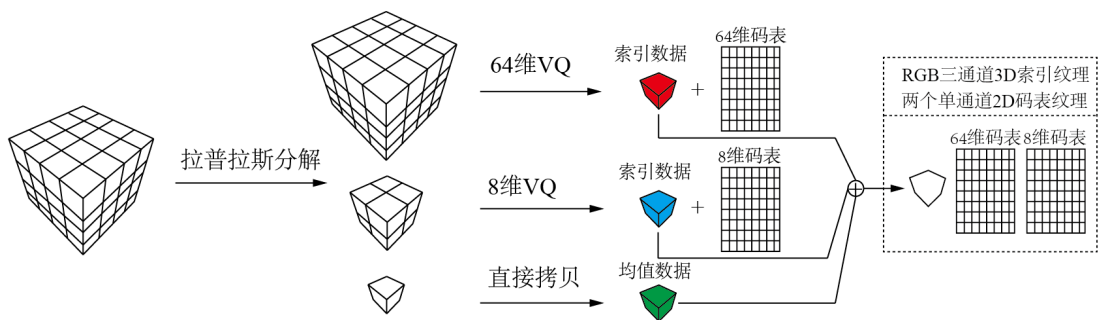


图 3 HVQ 压缩端流程图

图 4 是基于 GPU 的 HVQ 压缩域体绘制流程图

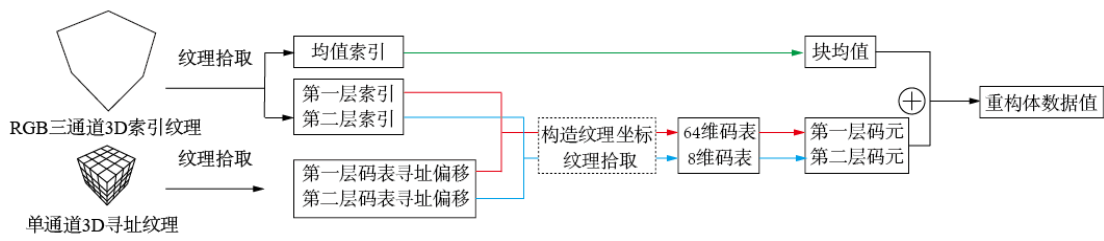
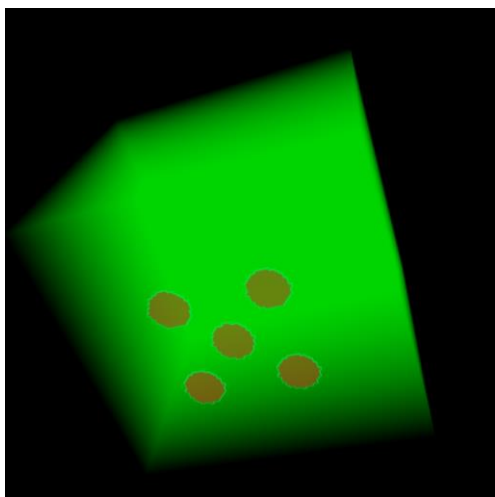


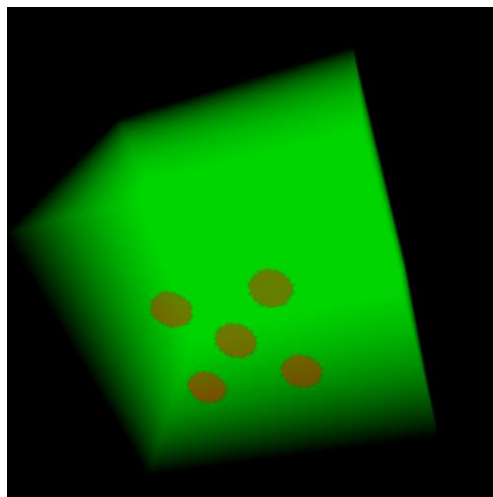
图 4 GPU HVQ 压缩域体绘制流程图

目前的主要问题在于 GPU 端实现 PSH 的解码工作没能很顺利地解决，从而导致结果没法按我们的方法得到。但是，在 CPU 端的 PSH 解码没有问题，结果如图 5 和 6 所示。

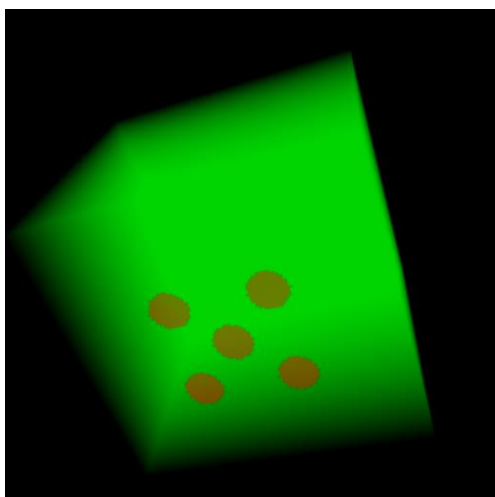
图 5a 是 5jets\_energy 第一帧的直接体绘制结果，图 5b 是对应的 HVQ CDVR 结果，图 5c 是本文方法的结果，图 5d 是图 5a 和图 5b 的差异图，在该数据中，由于残差数据全为 0，因此 HVQ CDVR 的结果和本文方法的结果一致，图 5b 和图 5c 相同。



(a)原始结果



(b)HVQ CDVR 结果

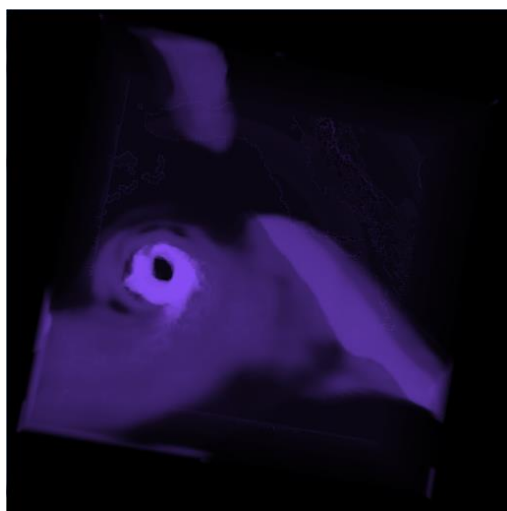


(c)PSH HVQ CDVR 结果

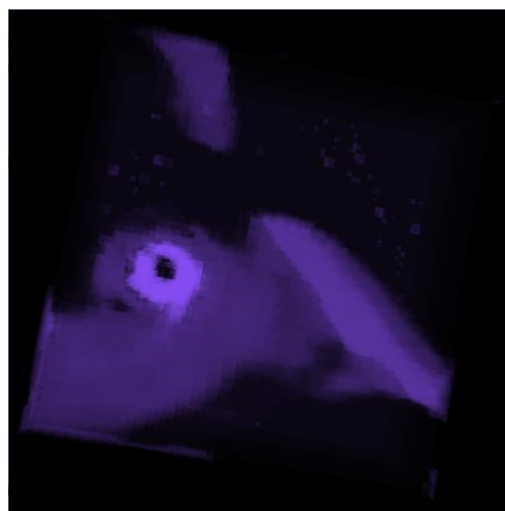


(d) a 与 b 差异图

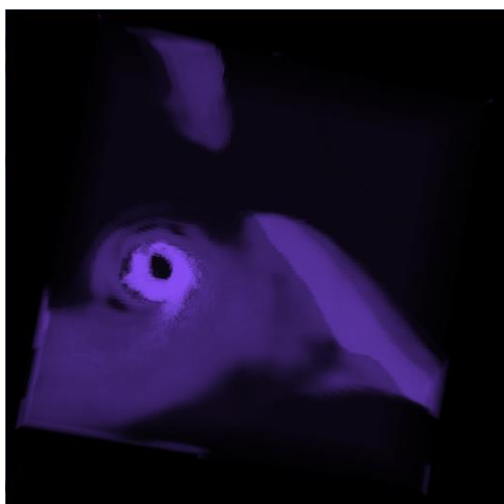
图 5 数据 5jets\_energy 第一帧绘制结果



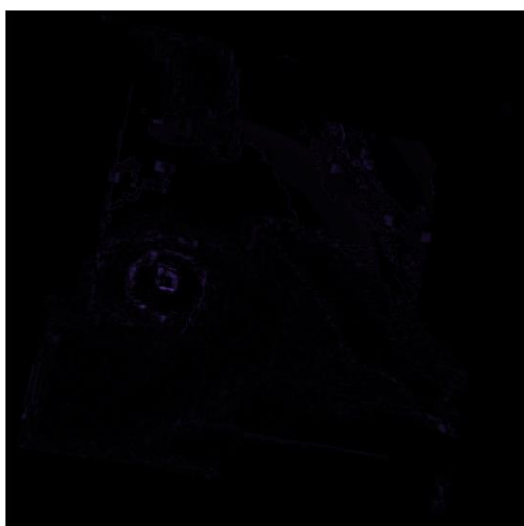
(a)原始结果



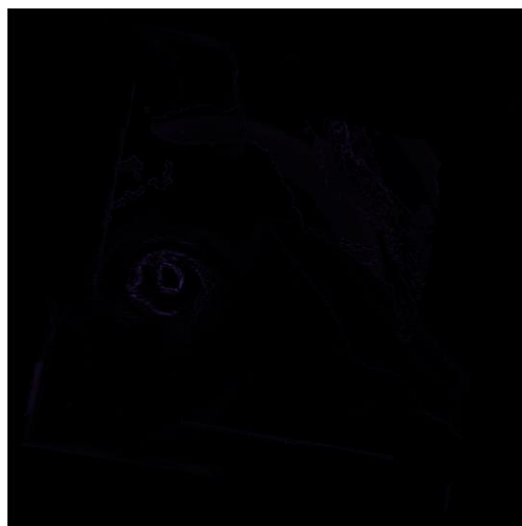
(b)HVQ CDVR 结果



(c)PSH HVQ CDVR 结果



(d) a 与 b 差异图



(e) a 与 c 差异图

图 6 数据 VISContest2004 飓风云场 cloud 第一帧绘制结果

图 6a 是 VISContest2004 飓风云场 cloud 第一帧的直接体绘制结果，图 6b 是对应的 HVQ CDVR 结果，图 6c 是本文方法的结果，图 6d 是图 6a 和图 6b 的差异图，图 6e 是图 6a 和图 6c 的差异图，在该数据中，由于有残差数据，因此 HVQ CDVR 的结果和本文方法的结果不一致，我们的方法解码绘制的结果与原始体数据更接近，误差更小。

上述结果的 PSH CDVR 是在 CPU 端做出来的，理论上和效果上都和 GPU 的应该是一致的。接下来，希望可以尽快调出 GPU 的版本，和流水线保持一致，完成论文。

除了完成这篇论文的结果实现，游戏教材的整理、修改以及 vis paper collection 工作都将同步展开。气象工作和朱标保持沟通，交接。